🔥 Closures 🔥

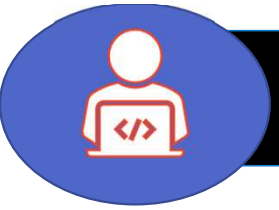
(IIFE)

...Spread

తెలుగు లో JavaScript

Venkatesh Mogili

#WebGuru



Closures

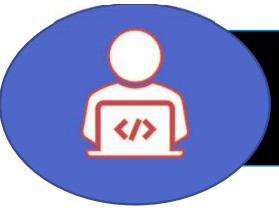
- A closure is a feature in JavaScript where an inner function has access to the outer function's variables and parameters — a scope chain.
- ✓ **the inner function remembers the environment in which it was created.**

The closure has three scope chains:

- ✓ It has access to its own scope — variables defined between its curly brackets.
- ✓ It has access to the outer function's variables
- ✓ It has access to the global variables

Real time example

Closure



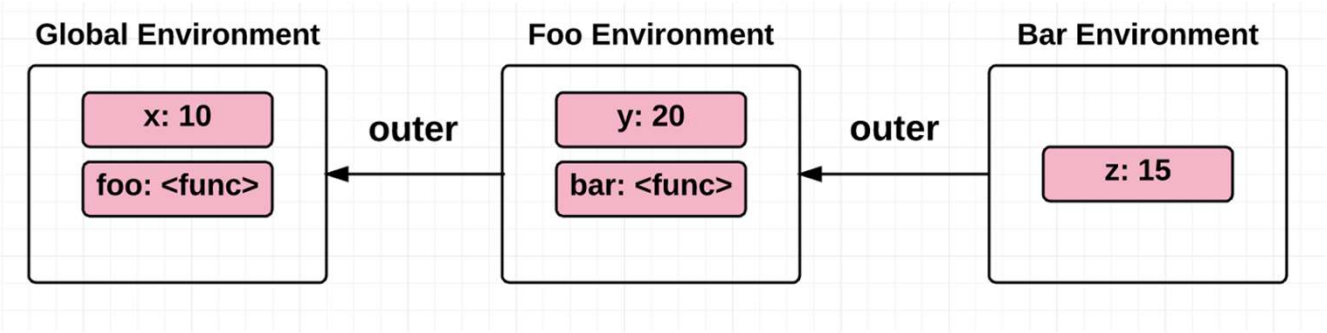
Scope Chain

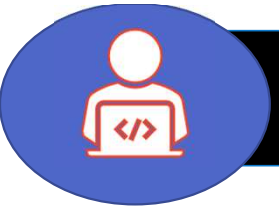
- This set of identifiers that each environment has access to is called *"scope."*
- We can nest scopes into a hierarchical chain of environments known as the *"scope chain"*.

```
Global Execution Context
1  var x = 10;
2
3  function foo() {
4              Execution Context (foo)
5    var y = 20; // free variable
6
7    function bar() {
8                Execution Context (bar)
9      var z = 15; // free variable
10     var output = x + y + z;
11     return output;
12   }
13
14   return bar;
15 }
```

Scope Chain

```
var x = 10;

function foo() {
  var y = 20; // free variable
  function bar() {
    var z = 15; // free variable
    return x + y + z;
  }
  return bar;
}
```





IIFE

1. An **IIFE** (Immediately Invoked **Function** Expression) is a JavaScript **function** that runs as soon as it is defined.
2. They don't pollute the global object, and they are a simple way to isolate variables declarations.

IIFE Function

```
var username='Ve
(function(userna
  username = 'Vi
})(username)
console.log(user
```

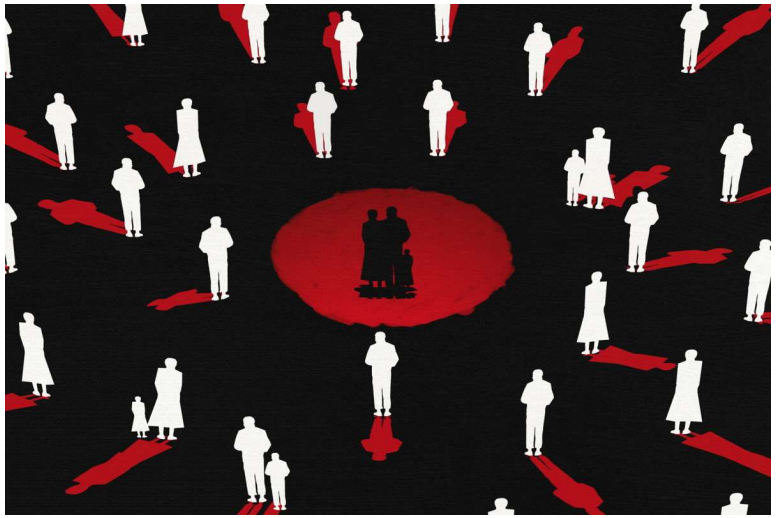
l Function:

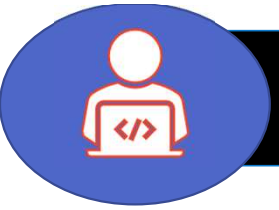
```
ame='Venkatesh';
a(){
e = 'Vivek'


og(username);
```

Spread

1. Spread syntax (...) allows an iterable such as an array expression (OR) string to be expanded in places where zero or more arguments are expected, (OR) an object expression to be expanded in places where zero or more key-value pairs are expected.

```
// Merging two arrays
let arr1 =[1,2,3];
let arr2 =[4,5,6];
let arr3 = [...arr1,...arr2];
//output: [1,2,3,4,5,6]
```





Summary

1. A closure is a feature in JavaScript where an inner function has access to the outer function's variables and parameters.

2. An **IIFE** (Immediately Invoked **Function** Expression) is a JavaScript **function** that runs as soon as it is defined and it won't pollute the global objects.

3. Spread syntax (...) allows an iterable to spread arrays, strings, objects.

Don't forget to Share and Subscribe

LIKE SHARE COMMENT

SUBSCRIBE

Thank you for watching

Subscribe

and

Stay Tuned

For more updates...

VENKATESH MOGILI

#Web Guru